

## **METHODS AND SYSTEMS FOR MOBILE DEVICE MESSAGING**

### **Technical Field**

The present invention relates generally to the field of mobile computing devices and more  
5 particularly to sending messages to a mobile device through a web service.

### **Background of the Invention**

A wide variety of mobile computing devices such as cellular telephones, pagers, Personal  
Digital Assistants (PDAs), and others are commonly in use. Such devices may be connected with  
10 a wireless network such as a cellular network through which the mobile devices may connect  
with other computing devices and other mobile devices. For example, one mobile device may  
send a voice or text message over a wireless network to another wireless device.

Additionally, fixed networks such as the Internet and other types of Wide Area Networks  
(WANs) and Local Area Networks (LANs) continue to develop. Attempts have been made to  
15 bridge wireless networks to fixed networks in limited ways. For example, some wireless devices  
include browser software for surfing or browsing the Internet. Additionally, email and text  
messages may be sent from fixed networks to various wireless devices.

In some cases, a message, such as an email or other text message, may be sent to a mobile  
device in the form of a HyperText Mark-up Language (HTML) file using the Hyper Text Transfer  
20 Protocol (HTTP). For example, a user of a mobile device or other computing device may send an  
email to a user of a mobile device in the form of HTML text via a server with which the mobile  
device is connected. The server may then forward the message to the mobile device.

In other cases, an application may send information to a mobile device by sending an  
email to the mobile device using normal email protocols. For example, a user of an application  
25 on a mobile device or other computing device, while executing an application, may initiate the  
sending of an email including some content information. The email is generated by the

application or another email program and is then sent to a server connected with the initiating device using standard email protocols. The mail server then forwards the email message to the intended recipient.

However, both of these approaches present some problems and limitations. First of all, transferring HTML messages to mobile devices is not widely supported, either by mobile devices or by application programs which may originate such messages. For example, a personal organizer program such as Microsoft Outlook® may not support the generation of HTML text to transfer a calendar appointment or task reminder to a mobile device. Additionally, an HTML file for transferring such information is highly platform or application specific. Therefore, this method presents compatibility problems between various systems and applications. Further, various security features, such as corporate firewalls, proxy servers, etc., limit the types of messages that may be transferred out of or through a given fixed network. Therefore, the sending of an HTML or email message to an unknown or unrecognized device will be blocked. This limits or complicates the use of these methods with some fixed networks. It is with respect to these considerations and others that the present invention has been made.

### **Summary of the Invention**

In accordance with the present invention, the above and other problems are solved by methods and systems for mobile device messaging. These methods and systems include a web service client that converts content to be sent to a mobile device to a form readable by a web service. The client generates and sends one or more short messages to the web service containing the converted content data. The web service receives the short messages, processes the messages, and converts the content data to a form that is readable by the intended mobile device. The web service then forwards the content data to a wireless network operator for delivery to the mobile device.

In accordance with other aspects of the present invention, a method of mobile device messaging comprises collecting information from an originating system that includes content data to be sent to the mobile device. One or more short messages are then generated for encapsulating the content data. The one or more short messages are formatted to be readable by a web service  
5 and the content data is formatted to be readable by the wireless device. Next, the one or more short messages are sent to the web service for delivery to the mobile device.

According to other aspects of the present invention, a system for mobile device messaging comprises a processor and a memory coupled with, and readable by, the processor. The memory contains instructions that, when executed by the processor, cause the processor to collect from an  
10 originating system information including content data to be sent to the mobile device. One or more short messages are generated for encapsulating the content data. The one or more short messages are formatted to be readable by a web service and the content data is formatted to be readable by the wireless device. The one or more short messages are sent to a web service for delivery to the mobile device.

15 According to yet another aspect of the present invention, a system for mobile device messaging comprises a processor and a memory coupled with and readable by the processor. The memory contains a series of instructions that, when executed by the processor, cause the processor to receive a short message from a web service client. The short messaging is formatted to be readable by a web service and contains content data formatted to be readable by a mobile  
20 device. A determination is made as to whether a sender of the short message is authentic and authorized to send the short message. If the sender of the short message is authentic and authorized to send the short message, the content data is sent to the mobile device.

The invention may be implemented as a computer process, a computing system, or as an article of manufacture such as a computer program product or computer readable media. The  
25 computer program product may be a computer storage media readable by a computer system and encoding a computer program of instructions for executing a computer process. The computer

program product may also be a propagated signal on a carrier readable by a computing system and encoding a computer program of instructions for executing a computer process.

These and various other features as well as advantages, which characterize the present invention, will be apparent from a reading of the following detailed description and a review of  
5 the associated drawings.

### **Brief Description of the Drawings**

FIG. 1 illustrates an environment that includes a system for sending messages to a mobile device according to an embodiment of the present invention.

10 FIG. 2 illustrates an example of a suitable computing system environment on which embodiments of the invention may be implemented.

FIG. 3 illustrates functional components of a system for sending messages to a mobile device according to an embodiment of the present invention.

FIG. 4 illustrates an exemplary data format for a short message from a client system or  
15 mobile device to a web service according to an embodiment of the present invention.

FIG. 5 illustrates an exemplary data format for a response from a web service to a client system or mobile device according to an embodiment of the present invention.

FIG. 6 is a flowchart illustrating, at a high level, sending a message to a mobile device according to an embodiment of the present invention.

20 FIG. 7 is a flowchart illustrating generating and sending a message to a mobile device according to an embodiment of the present invention.

FIG. 8 is a flowchart illustrating handling a message to a mobile device according to an embodiment of the present invention.

FIG. 9 is a flowchart illustrating handling a response from a web service according to an  
25 embodiment of the present invention.

### **Detailed Description of the Invention**

Before describing various embodiments of the present invention, some terms that will be used throughout this description will be defined.

“Mobile messaging” refers to sending and/or receiving data, such as text messages, email, reminders, calendar appointments, video, audio, graphics, and other types of data, to or from a mobile device over a wireless network.

A “Multimedia Message Service (MMS)” is a service for sending and receiving graphics, video, sounds and other multimedia content over a network that is widely supported by various mobile devices.

A “Multimedia Message Service Center (MMSC)” is a system typically operated by a wireless network operator for receiving MMS messages and directing the messages to an intended recipient.

A “short message” is a message of a limited size and pre-defined format readable by a web service and used to encapsulate data transferred to or from a mobile device.

A “Short Message Service (SMS)” is a service for sending and receiving short text messages over a network that is widely supported by various mobile devices.

A “Short Message Service Center (SMSC)” is a system typically operated by a wireless network operator for receiving SMS messages and directing the messages to an intended recipient.

“Simple Object Access Protocol (SOAP)” is a messaging protocol using eXtensible Markup Language (XML) to encode information in web service requests and responses.

A “web service” is a set of self-contained, modular applications or services stored on a server and made available for access across the Internet. A web service provides the functionality of the various applications stored thereon to a client application without requiring the client application to provide that functionality. To use the services, the client invokes the service and/or passes data to the web service in a form readable by that service.

FIG. 1 illustrates a system for sending messages to a mobile device according to an embodiment of the present invention. The system **100** includes two web services **115** and **145**, two client systems **105** and **135**, two wireless network operator systems **120** and **150** including a Short Message Service Center (SMSC) and/or a Multimedia Message Service Center (MMSC),  
5 and a number of mobile devices **125**, **130**, **155**, and **160**. In actual implementation, any number of web services, client systems, wireless network operator systems, and mobile devices may be used.

The client systems **105** and **135** may be connected with one or more of the web services **115** and **145** via the Internet **110** or other network. The provider of a web service **115** may be  
10 either integrated with or separate from the wireless network operator. When separate, the web service **115** may connect with the wireless network operator system **120** via the Internet **110** or other network. Alternatively, the provider of a web service **145** may be the same entity that provides the wireless service and therefore also maintains the wireless network operator system **150**. In this case, the web service **145** may be connected with the wireless network operator  
15 system **150** via an Intranet **165** or other type of network. As will be seen, the web services **115** and **145** provide a set of modular applications for transferring content from the client systems **105** and/or **135** to one or more of the mobile devices **125**, **130**, **155**, and/or **160** through a pre-defined interface. In this way, to send a message to a mobile device, the client system can present the content data to the web service without a need to perform additional functions associated with  
20 this transfer.

A user of the client system **105** or **135** may initiate the sending of a message to one or more of the mobile devices **125**, **130**, **155**, and **160**. For example, the user of a client system **105** or **135**, while viewing an email message, may choose to forward that message to a mobile device **125**, **130**, **155**, and/or **160**. The client system **105** or **135** then generates one or more short  
25 messages to encapsulate the content of the email being forwarded. That is, the client system

generates a message that is readable by the web service **115** or **145**. The short message may include the content data in the form of a Short Message Service (SMS) message or a Multimedia Message Service (MMS) message or similar format that will be readable by the mobile device to which the content is being sent. The short message encapsulating the content follows a schema or format similar to that described below with reference to FIG. 4. Once the short messages are generated, the client system **105** or **135** sends the short messages to the web service **115** or **145** via a channel established with the web service **115** or **145** over the Internet **110**. In one example, the short message may be conveniently sent to the web service **115** or **145** using the widely supported Simple Object Access Protocol (SOAP).

The web service **115** or **145**, as will be described in detail below, parses the short message, checks the authenticity and authority of the user of the client system **105** or **135** to send the short message, and if the user is authenticated and authorized, sends the content of the short message to the wireless network operator system **120** or **150**. Additionally, the web service **115** or **145** may perform other functions such as checking the short message for errors, logging the message and results of sending the message, etc. In some cases, the web service **115** or **145** may even modify the format of the content contained in the short message to place it into a format readable by a specific mobile device.

In the case where the web service **115** and wireless network operator system **120** are operated by separate entities, the wireless operator system **120** includes an SMSC gateway (not shown) and/or MMSC gateway (not shown) for receiving and handling the short message sent by the web service **115** and forwarding the message to the SMSC and/or MMSC. If, however the web service **145** and wireless operator system **150** are maintained by the same entity, the web service may pass the short message directly to the SMSC and/or MMSC and the wireless network operator system **150**.

The successfully processed short message will be put in the queue of the SMSC or MMSC by the wireless network operator system **120** or **150** which then sends a response to the

web service **115** or **145** informing the web service **115** or **145** of the status of the message delivery. That is, the wireless network operator system **120** or **150** informs the web service **115** or **145** of whether the message was successfully delivered to the SMSC or MMSC queue or, if not successfully delivered, may return error information. The wireless network operator system **120** or **150** then sends the message to the mobile device designated to be the recipient.

The web service **115** or **145** then, based on the response from the wireless network operator system **120** or **150**, generates a response to the short message from the client system **105** or **135** and sends the response to the client. That is, the web service **115** or **145** will generate a response message based on the response from the wireless network operator **120** or **150**. The format of the response message will be discussed in detail below but generally may include an indication of success or failure of delivery of the message, return codes, error codes, or other information. The web service **115** or **145** then sends the response to the client system **105** or **135** which may in turn inform the user of the response.

FIG. 2 illustrates an example of a suitable computing system environment on which embodiments of the invention may be implemented. This system **200** is representative of one that may be used to serve as a client system or a server providing the web service. In its most basic configuration, system **200** typically includes at least one processing unit **202** and memory **204**. Depending on the exact configuration and type of computing device, memory **204** may be volatile (such as RAM), non-volatile (such as ROM, flash memory, etc.) or some combination of the two. This most basic configuration is illustrated in FIG. 2 by dashed line **206**. Additionally, system **200** may also have additional features/functionality. For example, device **200** may also include additional storage (removable and/or non-removable) including, but not limited to, magnetic or optical disks or tape. Such additional storage is illustrated in FIG. 2 by removable storage **208** and non-removable storage **210**. Computer storage media includes volatile and nonvolatile, removable and non-removable media implemented in any method or technology for



storage of information such as computer readable instructions, data structures, program modules or other data. Memory **204**, removable storage **208** and non-removable storage **210** are all examples of computer storage media. Computer storage media includes, but is not limited to, RAM, ROM, EEPROM, flash memory or other memory technology, CD-ROM, digital versatile  
5 disks (DVD) or other optical storage, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other medium which can be used to store the desired information and which can be accessed by system **200**. Any such computer storage media may be part of system **200**.

System **200** typically includes communications connection(s) **212** that allow the system to  
10 communicate with other devices. Communications connection(s) **212** is an example of communication media. Communication media typically embodies computer readable instructions, data structures, program modules or other data in a modulated data signal such as a carrier wave or other transport mechanism and includes any information delivery media. The term “modulated data signal” means a signal that has one or more of its characteristics set or  
15 changed in such a manner as to encode information in the signal. By way of example, and not limitation, communication media includes wired media such as a wired network or direct-wired connection, and wireless media such as acoustic, RF, infrared and other wireless media. The term computer readable media as used herein includes both storage media and communication media.

20 System **200** may also have input device(s) **214** such as keyboard, mouse, pen, voice input device, touch input device, etc. Output device(s) **216** such as a display, speakers, printer, etc. may also be included. All these devices are well known in the art and need not be discussed at length here.

A computing device, such as system **200**, typically includes at least some form of  
25 computer-readable media. Computer readable media can be any available media that can be

accessed by the system **200**. By way of example, and not limitation, computer-readable media might comprise computer storage media and communication media.

FIG. 3 illustrates functional components of a system for sending messages to a mobile device according to an embodiment of the present invention. This example includes a client

5 system **305** connected with a web service **310** via the Internet **315** or other network.

Alternatively, client system **305** may be a mobile device initiating a message to another mobile device. In such a case, the components and operations will be the same as discussed below with reference to the client system **305**.

The client system **305** includes messaging application **320** such as Microsoft Outlook® or  
10 other application program and a web service client **325**. During execution, the application **320** may initiate sending of a message to a mobile device such as through a user selecting a user interface element. For example, a user of Microsoft Outlook® may wish to send a calendar appointment to his co-worker's PDA or cell phone on a separate platform or that uses a separate message application. In such a case, the user may select a user interface element indicating an  
15 option to send a message to a mobile device. In response, the application **320** then invokes, instantiates, or otherwise triggers the web service client module **325**.

The web service client **325** includes a user interface module **340**, a message generation module **330**, and a web service communication module **326**. The user interface module **340** queries the user for specific information to be included in the message. For example, the user  
20 interface module **340** prompts the user for destination information such as the destination device's service provider, phone number and/or other identifying information. Additionally, the user interface module **340** may query the user for information such as a user identification and/or password to be used by the web service **310** for authentication and/or authorization of the message as will be discussed below. In alternative embodiments, the user information may be  
25 preset and retrieved as needed to allow for a more automatic process of message generation.

Generally speaking, the content to be sent to the destination mobile device is placed into a form that is readable by the destination mobile device. For example, since SMS and MMS are widely supported by most mobile devices, these formats are convenient for transferring the content data. Additionally, since a message such as an SMS message or MMS message has a pre-defined size limit, the content is checked against this limit. If the size of the content exceeds the size limit for the content, the message generation module **330** divides the content into a number of segments to be encapsulated in more than one short message. This division or split may be made by default. In other cases, the user may be queried to determine whether the user agrees to allow the division of data into multiple messages.

The message generation module **330** then generates a short message readable by the web service to encapsulate the content to be sent to the destination mobile device. For example, an XML message may be generated following a pre-defined schema. Details of such a short message and an exemplary schema will be discussed below with reference to FIG. 4. Alternatively, another format or schema, readable by the web service may be used.

Once the short message is generated the web service communication module **326** sends the short message **345** to the web service **310** via a channel established over the Internet **315**. In one example, the short message may be conveniently sent to the web service **310** using the widely supported Simple Object Access Protocol (SOAP).

The web service **310** includes a Web Service Description Language (WSDL) file **335** defining a web service module **350**. That is, the WSDL file **335** may include an XML or other description of the services provided by the web service module **350**. An exemplary WSDL file is listed below in Table 1.

**Table 1:**

```
<?xml version="1.0" encoding="utf-8"?>
<definitions xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:s="http://www.w3.org/2001/XMLSchema"
```

```
xmlns:s0="urn:Microsoft.Office.OMMS.OMMWS" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/" xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
targetNamespace="urn:Microsoft.Office.OMMS.OMMWS" xmlns="http://schemas.xmlsoap.org/wsdl/">
<types>
5   <s:schema elementFormDefault="qualified" targetNamespace="urn:Microsoft.Office.OMMS.OMMWS">
      <s:element name="sendXml">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="SMSData" type="s:string" />
10         </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="sendXmlResponse">
        <s:complexType>
15         <s:sequence>
          <s:element minOccurs="0" maxOccurs="1" name="sendXmlResult" type="s:string" />
        </s:sequence>
        </s:complexType>
      </s:element>
20    </s:schema>
  </types>
  <message name="sendXmlSoapIn">
    <part name="parameters" element="s0:sendXml" />
  </message>
25  <message name="sendXmlSoapOut">
    <part name="parameters" element="s0:sendXmlResponse" />
  </message>
  <portType name="OMMWebServiceSoap">
    <operation name="sendXml">
30      <documentation>This method handles the request from Microsoft Office Mobile Message Add-
in.</documentation>
      <input message="s0:sendXmlSoapIn" />
      <output message="s0:sendXmlSoapOut" />
    </operation>
35  </portType>
  <binding name="OMMWebServiceSoap" type="s0:OMMWebServiceSoap">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document" />
    <operation name="sendXml">
      <soap:operation soapAction="urn:Microsoft.Office.OMMS.OMMWS/sendXml" style="document" />
40    <input>
      <soap:body use="literal" />
    </input>
    <output>
      <soap:body use="literal" />
45    </output>
    </operation>
  </binding>
  <service name="OMMWebService">
    <documentation>Microsoft Office Mobile Message Web Service.</documentation>
50    <port name="OMMWebServiceSoap" binding="s0:OMMWebServiceSoap">
      <soap:address location="http://localhost/OMMWS/OMMWebService.asmx" />
    </port>
  </service>
</definitions>
55
```

The web service module **350** receives the short message **345** from the client system **305**.

The web service module **350** then parses the short message and passes the sender's identification

information to the authentication module 370. The authentication module 370 determines whether the user of the client system 305 is who he claims to be and is authorized to send the message. This determination may be made by comparing the sender's information such as an identification and password against information in the subscriber database 375. If the provider of the web service 310 is different from the provider of the wireless service, the subscriber database 375 may be maintained by the wireless service provider rather than the provider of the web service 310. In such a case, the authentication module 370 may request the subscriber information from the SMSC or MMSC of the wireless network operator.

If the user of the client system 305 is authentic and authorized, the web service module 350 passes the content of the short message to the communication module 365. The communication module 365 then sends the message to the SMSC of the wireless network operator. As discussed above, the wireless network operator sends back a response indicating success or failure of delivery of the message. The communication module 365 receives this response and passes it along to the web service module 350. The web service module 350 in turn generates a response to the short message 345 from the client system 305 based on the response from the wireless network operator system. That is, the web service module 350 generates a response based on the response from the wireless network operator. The format of the response will be discussed in detail below with reference to FIG. 5 but generally may include an indication of success or failure of delivery of the message, return codes, error codes, or other information. The web service module 350 then sends the response 380 to the client system 305 via the Internet 315. Additionally, tracking/logging module 360 may record information related to the short message 345 and the response 380 such as the sender's identification, time, results, error codes, etc.

The web service communication module 325 of the client system 305 receives the response 380 and may parse the response and pass the results to the user interface module 340.

The user interface module 340 may then inform the user of the client system 305 of the success or failure of the message. For example, the user interface module may open a window or other user interface element to display a message indicating success or failure of the message. A message indicating failure may also indicate error codes or messages.

5           FIG. 4 illustrates an exemplary format for a message from a client system or mobile device to a web service according to an embodiment of the present invention. This example illustrates one possible format for a short message. However, depending upon the exact implementation, the format may vary. For example, additional elements may be included, the elements may be in a different order or some illustrated elements may be omitted. This message  
10       format may be defined as an XML schema or in another similar manner. Table 2 is an exemplary XML schema for defining a message format as illustrated in FIG. 4.

**Table 2:**

```
15 <?xml version="1.0" encoding="utf-8" ?>
<xs:schema targetNamespace="urn:Microsoft.Office.OMMS.SMSData" elementFormDefault="qualified"
xmlns="urn:Microsoft.Office.OMMS.SMSData" xmlns:mstns="urn:Microsoft.Office.OMMS.SMSData"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="SMSData">
    <xs:complexType>
      <xs:sequence>
20        <xs:element name="Carrier" type="xs:string" minOccurs="1" maxOccurs="1" />
        <xs:element name="Id" type="xs:string" maxOccurs="1" minOccurs="1" />
        <xs:element name="Password" type="xs:string" maxOccurs="1" minOccurs="1" />
        <xs:element name="ToMobile" type="xs:string" maxOccurs="1" minOccurs="1" />
25        <xs:element name="Message" type="xs:string" minOccurs="1" maxOccurs="unbounded" />
        <xs:element name="MsgType" type="xs:string" minOccurs="1" maxOccurs="1" />
        <xs:element name="Scheduled" type="xs:time" minOccurs="0" maxOccurs="1" />
        <xs:element name="MMS" minOccurs="0" maxOccurs="1">
          <xs:complexType>
30            <xs:simpleContent>
              <xs:extension base="xs:boolean">
                <xs:attribute name="format" type="xs:string" default="text" use="optional" />
              </xs:extension>
            </xs:simpleContent>
35          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
40 </xs:schema>
```

The message format illustrated in FIG. 4 includes a service provider element **405**, a sender's ID element **410**, a password element **415**, a destination element **420**, a message filed **425**, a message type element **430**, a schedule element **435**, and a message format element **440**.

The service provider element **405**, identified by the name "Carrier" in Table 2, indicates  
5 the service provider that sets up the web service. Depending on whether the service provider is also a wireless service carrier, it may communicate with an SMSC or MMSC or gateway of a cooperative carrier of the wireless service as discussed above with reference to FIG. 1.

The sender's ID element **410** indicates the user name or other identifying information about the sender of the message. The sender's ID element **410** is identified by the name "Id" in  
10 Table 2. This information may be used by both the web service and the destination mobile device to identify the origin of the message. For example, the sender's ID element **410** may be used by the web service to track or log messages or to determine whether the sender is authorized to send the message.

The password element **415**, identified by the name "Password" in Table 2, indicates a  
15 password for the sender. This password may be used, for example, by the web service to identify and authenticate the sender of the message.

The destination element **420** indicates the phone number, address or other identification of the destination mobile device. The destination element **420** is identified by the name  
"ToMobile" in Table 2. This information is passed from the web service to the wireless network  
20 service operator for delivery of the message to the mobile device.

The message element **425**, identified by the name "Message" in Table 2, contains the content of the message. For example, the message element **425** may contain the text of an email, a calendar appointment, a task reminder, or other type of content. In some cases, where the content being sent exceeds the predetermined size of the small message, the message element **425**  
25 may contain a portion of a larger content as discussed above.

The message type element **430** indicates the type of content being delivered. For example, the message type element **430** may indicate that the data in the message element **425** is an email message or a calendar appointment. The message type element **430** is identified by the name "MsgType" in Table 2.

5           The schedule element **435** may indicate a specific time for delivery of the message to the destination mobile device. The schedule element **435** is identified by the "Scheduled" name in Table 2. This information may indicate, for example, a time and date at which the web service should deliver the message to the wireless network service operator for delivery to the destination mobile device.

10           The message format element **440**, indicated by the "MMS" name in Table 2, indicates the type of message contained in the message element **425**. For example, the message format element **440** may indicate whether the message element **425** is an SMS message or an MMS message.

FIG. 5 illustrates an exemplary format for a response from a web service to a client  
15   system or mobile device according to an embodiment of the present invention. This example illustrates one possible format for a response to a short message. However, depending upon the exact implementation, the format may vary. For example, additional elements may be included, the elements may be in a different order or some illustrated elements may be omitted. This response format may be defined as an XML schema or in another similar manner. Table 3 is an  
20   exemplary XML schema for defining a response format as illustrated in FIG. 5.

**Table 3:**

```
<?xml version="1.0" encoding="utf-8" ?>
<xs:schema targetNamespace="urn:Microsoft.Office.OMMS.Response" elementFormDefault="qualified"
25   xmlns="urn:Microsoft.Office.OMMS.Response" xmlns:mstns="urn:Microsoft.Office.OMMS.Response"
   xmlns:xs="http://www.w3.org/2001/XMLSchema" id="Response">
   <xs:element name="Response">
      <xs:complexType>
          <xs:sequence>
30           <xs:element name="Result" minOccurs="1" maxOccurs="unbounded">
              <xs:complexType>
```



```

5      <xs:sequence>
        <xs:element name="ToMobile" minOccurs="0" maxOccurs="1">
          <xs:complexType mixed="true">
            <xs:sequence>
              <xs:element name="Count" type="xs:string" minOccurs="0"
maxOccurs="1" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
10      <xs:element name="RetCode" type="xs:string" maxOccurs="1" minOccurs="1" />
        <xs:element name="ErrCause" type="xs:string" maxOccurs="1" minOccurs="1" />
        <xs:element name="Message" type="xs:string" maxOccurs="1" minOccurs="1" />
      </xs:sequence>
    </xs:complexType>
15  </xs:element>
  </xs:sequence>
</xs:complexType>
20 </xs:element>
</xs:schema>

```

The response format illustrated in FIG. 5 includes one or more result elements **505** and **535**. Each result element may include a number of child elements representing details of the sending result. In this example, the result element **505** and **535** includes an optional recipient element **510**, an optional count element **515** that is a child of the recipient element **510** if present, a return code element **520**, an error cause element **525**, and a message element **530**.

The result element **505** or **535**, identified by the name “Result” in Table 3, indicates a set of processing results having the same return code. That is, the result element **505** indicates the wireless network service operator’s results of sending the message to the destination device. Additionally, the result element **505** may indicate results of checks performed by the web service such as authenticity and authorization checks for the message. Multiple result elements may be used in the case of multiple or split messages. For example, if two short messages were sent, one resulting in errors, two result elements may be included in the response. In this case, one result element may indicate the successful result while the other result element indicates the error result.

The optional recipient element **510** indicates the cell phone number, address, or other identifying information for the destination mobile device. The recipient element **510** is identified by the “ToMobile” name in Table 3. The recipient element **510** may be omitted generally indicating there is an error and the error has nothing to do with the recipient. For example, if the

sender gives an invalid password, the resulting error has nothing to do with the recipient.

Therefore, the recipient element **510** may be omitted from the response **500** since it is not relevant to that error.

The optional count element **515** that is a child of recipient element **510**, identified by the  
5 “Count” name in Table 3, indicates the number of messages to which the recipient element **505**  
corresponds in the event of split or multiple messages. For example, the count element may  
indicate three meaning that three messages were delivered with the results indicated by the  
result element **505**. To further illustrate, consider an example where a client system sends a  
group of five messages to one mobile device. If three messages are successfully delivered and  
10 two messages fail because the messages were not authorized, two results will be returned. One  
result will indicate a successful result with a count of 3 and the other result will indicate an  
unsuccessful result with a count of 2.

The return code element **520** may contain a numeric value or other code indicating the  
result of the web service’s handling of the message to which the response corresponds. For  
15 example, the return code element **520** may indicate web service errors, SMSC errors,  
authentication errors and others. The return code element **520** is identified by the “RetCode”  
name in Table 3.

The error cause element **525**, identified by the “ErrCause” name in Table 3, may contain  
an indication of the component that caused an error. For example, if the user’s password in the  
20 original short message was invalid, the error cause element **525** can indicate the Client as the  
cause of the error. In another example, if the SMSC of the service provider returns a response to  
the web service indicating failure of the delivery of the message, the error cause element **525** may  
indicate the SMSC as the cause.

The message element **530**, identified by the “Message” name in Table 3, may contain a  
25 message describing the error. For example, the message element **530** may contain text or other  
easily readable information describing the nature of or reason for the error. This message may be

displayed to the user of the client system or initiating mobile device to inform the user of the nature of the error.

More than one set of result elements in a response generally indicates that there is more than one group of results having the same return code. Each successive result element follows the same format such as the exemplary format described above. To further illustrate, consider an example where a client system sends the same message to three mobile devices at the same time. If two messages are successfully delivered to each of the first two recipients and one message fails because the recipient's phone number is incorrect, two results will be returned. One result will indicate a successful result with a success return code and the other result will indicate an unsuccessful result with invalid recipient number return code.

The logical operations of the various embodiments of the present invention are implemented (1) as a sequence of computer implemented acts or program modules running on a computing system and/or (2) as interconnected machine logic circuits or circuit modules within the computing system. The implementation is a matter of choice dependent on the performance requirements of the computing system implementing the invention. Accordingly, the logical operations making up the embodiments of the present invention described herein are referred to variously as operations, structural devices, acts or modules. It will be recognized by one skilled in the art that these operations, structural devices, acts and modules may be implemented in software, in firmware, in special purpose digital logic, and any combination thereof without deviating from the spirit and scope of the present invention as recited within the claims attached hereto.

FIG. 6 is a flowchart illustrating sending a message to a mobile device according to an embodiment of the present invention. This example refers to a client system initiating sending of the message to a mobile device where the client system may be a mobile device or not.

In this example, operation begins with gather operation 605 being performed by the client system. Gather operation 605 gathers the information to be included in the short message. For

example, a window or other user interface element may be opened to query the user for the destination mobile device's wireless service carrier, and number or address as well as the user's name or other identification and password. Additionally, the message type, delivery schedule, and message content may also be collected.

5           Next, generate operation **610** generates one or more short messages for encapsulating the content to be sent to the destination mobile device. For example, an XML file following the schema listed in Table 2 may be generated. As discussed above, if the content to be sent to the destination mobile device is longer than the predefined small message size, the content may be divided into smaller portions. In such a case, a small message for each portion will be generated.

10           Once one or more short messages have been generated, send operation **615** sends the short messages from the client system to the web service. This sending may be performed using Simple Object Access Protocol (SOAP) over a channel established between the client system and the web service over the Internet or other network.

          Receive operation **620** performed by the web service then receives the short messages  
15   from the client system. Then, query operation **625** determines whether to allow the messages. This determination may be based on a number of factors. For example, a first factor may be whether the user of the client system has provided a valid identification and password. Another factor may be whether the user of the client system is authorized to send messages based on having a current, paid account with the web service. Additionally, tests may be performed on the  
20   short message to determine whether it is well-formed and meets other possible requirements.

          If a determination is made at query operation **625** that the message should be allowed, send operation **630**, at the scheduled time or immediately if no scheduled time is indicated, sends the short message to the SMSC of the destination mobile device's service provider for delivery to the mobile device. Generate operation **635** then generates response such as an XML file  
25   following the schema listed in Table 3 above indicating failure of the message along with

appropriate return codes, error codes, and error messages and sends the response to the client system.

If a determination is made at query operation **625** that the message should not be allowed, generate operation **635** generates a response such as an XML file following the schema listed in Table 3 above indicating failure of the message along with appropriate return codes, error codes, and error messages and sends the response to the client system.

FIG. 7 is a flowchart illustrating generating and sending a short message to a mobile device according to an embodiment of the present invention. These operations may be initiated in multiple ways. First, generating and sending a short message may be initiated in an interactive mode wherein the user of the client system or initiating mobile device triggers the message through the user interface of the system or device. In another example, message generation and sending may be performed automatically upon the occurrence of some event or passage of some time or other condition. For example, a system may be configured to automatically forward all emails to a mobile device.

In the example illustrated in FIG. 7, operation begins with collect operation **705**. Collect operation **705** collects sender information such as the sender's user name or other identification, a password, and other possible information specific to the sender. Collection of this data may be performed by querying the user, by reading saved information previously provided by the user, or in other ways.

Next, collect operation **710** collects destination information. That is, collect operation **710** may open a window or other user interface element and query the user for destination information such as a service provider for the destination mobile device, a cell number for the destination mobile device, and other possible identifying information.

Collection operation **712** collects information such as the delivery time for the message. For example, the user may want the message to be delivered as soon as possible or at some later time, specified time. Therefore, the user may specify a time and date for delivery. Additionally,

other details of delivery may be collected by collection operation **712**. For example, the user may indicate whether to allow splitting of data that is longer than the short message size into multiple short messages for delivery. As with other data, this information may be pre-configured and retrieved automatically or may be collected by querying the user.

5           Collection operation **715** then collects the content to be sent to the destination mobile device encapsulated in one or more short messages. For example, collection operation **715** may read an email message, calendar appointment, word processing file, spreadsheet, or other information. The information to be sent may be identified by the user or may be assumed by the context in which generation of the message was initiated, such as clicking on a button or other  
10   user interface element while viewing the data.

          Query operation **720** then makes a determination as to whether the data to be sent should be included in one or more than one short message. For example, an SMS message has a pre-defined size limit. If the data is longer than this limit and a determination is made to split the message, the data is divided into more than one message. The determination to split the data may  
15   be based on delivery information collected in collection operation **712** described above. If, at query operation **720**, a determination is made that the data length is longer than the small message size and a user has chosen to split it, split operation **725** will divide the content into multiple portions, each small enough to be contained in a small message.

          Generate operation **730** then generates one or more small messages encapsulating the  
20   content data. That is, generate operation **730** generates one or more XML files following the schema listed in Table 2 and including the collected sender information, destination information, and content information.

          Finally, send operation **735** sends the one or more small messages to the web service. As discussed above, the small message may be sent to the web service using SOAP over a channel  
25   established between the client system or initiating mobile device and the web service over the Internet or other network.

FIG. 8 is a flowchart illustrating handling a short message to a mobile device according to an embodiment of the present invention. Here, operation begins with receive operation **805**. Receive operation **805** receives the short message from the client system or initiating mobile device.

5        Parse operation **810** then parses the received message. That is, the received message is parsed, based on the delimiters of the XML file to identify and read the individual elements as described above with reference to FIG. 4.

Query operation **815** then determines whether the user initiating the message is authentic and authorized to send the message. These determinations may be made by comparing the  
10    sender's information such as an identification and password against information in the subscriber database. For example, the sender's identification and password may be checked against recorded information. Additionally, a check of authority to send the message may be based on whether the sender has a current, paid account with the service provider. If the sender fails authentication or authorization, generate operation **820** generates a response message with the  
15    appropriate response code, return code, error code, error message, etc. Return operation **845** then sends the response back to the sender and log operation **850** records the message and results.

If, at query operation **815**, a determination is made that the sender is authentic and authorized, send operation **825** sends the short message to the SMSC of the wireless service provider as indicated in the short message.

20        Query operation **830** then checks the response from the SMSC to determine whether the message was successfully placed in the transmission queue of the SMSC. If the message was not successfully queued, generate operation **835** generates a response message with the appropriate response code, return code, error code, error message, etc. Return operation **845** then sends the response back to the sender and log operation **850** records the message and results.

25        If, at query operation **830** determines that the message was successfully queued, generate operation **840** generates a response message with the appropriate response code, return code,

error code, error message, etc. Return operation **845** then sends the response back to the sender and log operation **850** records the message and results.

FIG. 9 is a flowchart illustrating handling a response from a web service according to an embodiment of the present invention. In this example, operation begins with receive operation

5   **905**. Receive operation **905** receives the response from the web service.

Parse operation **910** then parses the response. That is, the received response is parsed, based on the delimiters of the XML file to identify and read the individual elements as described above with reference to FIG. 5.

Finally, optional notification operation **915** notifies the user of the success or failure of the  
10   message queued based on the response data. This operation is option since, in some cases, a notification may not be desired. For example, if the message was successfully placed into the transmission queue of the SMSC, no notification may be given. In other cases, all responses may prompt notification of the user. The notification may be in the form of opening a window or other user interface element to inform the user of the success of failure of the message. The  
15   notification may include return codes, error codes, and/or error messages from the response.

The various embodiments described above are provided by way of illustration only and should not be construed to limit the invention. Those skilled in the art will readily recognize various modifications and changes that may be made to the present invention without following the example embodiments and applications illustrated and described herein, and without  
20   departing from the true spirit and scope of the present invention, which is set forth in the following claims.